

AN IDENTIFICATION OF PRAGMATIC PRINCIPLES - FINAL REPORT

Document No.: TBD
Version/Revision: 0
Date: January 21, 1993
File: Principles Pragmatic Defoe 1993-0123 PrinWG.doc

Prepared by:
SE Principles Working Group
International Council on Systems Engineering (INCOSE)
2150 N. 107th Street, Suite 205
Seattle, WA 98133-9009

NOTICE

This data was prepared by the SE Principles Working Group of the International Council on Systems Engineering (INCOSE) for review and comment only. It has been released by that agency as INCOSE Technical Data. It is subject to change without notice and may not be referred to as an INCOSE Technical Product.

Copyright (c) 1993 by INCOSE, subject to the following restrictions:

Author Use. Authors have full rights to use their contributions in a totally unfettered way. Abstraction is permitted with credit to the source.

INCOSE Use. Permission to reproduce and use this document and to prepare derivative works from this document for INCOSE use is granted, provided this copyright notice is included with all reproductions and derivative works.

External Use. This document may not be shared or distributed to any non-INCOSE third party. Requests for permission to reproduce this document in whole or part, or to prepare derivative works of this document for external and/or commercial use are prohibited. Copying, scanning, retyping, or any other form of reproduction of the content of whole pages or source documents is prohibited and shall not be approved by INCOSE.

Electronic Version Use. Any electronic version of this document is to be used for personal use only and is not to be placed on a non-INCOSE sponsored server for general use. Any additional use of these materials must have written approval from INCOSE Central.

Permissions. INCOSE has granted permission to member companies of the INCOSE Corporate Advisory Board to post and use this document internally, subject to the external use restriction.

AN IDENTIFICATION OF PRAGMATIC PRINCIPLES - FINAL REPORT

Document No.: TBD
Version/Revision: 0
Date: January 21, 1993

Prepared by:

//signed//

J. C. DeFoe, Editor
SE Principles WG

January 21, 1993

Date

Approved by:

Chair, Technical Board,
Chair, Process & Improvements TC

_____ Date: _____

_____ Date: _____

Revision History

Revision	Revision Date	Change Description & Rationale
0	1/21/1993	Original

Preface

This document has been prepared and produced by a volunteer group of contributors within the International Council on Systems Engineering (INCOSE): the SE Practices Working Group, Subgroup on Pragmatic Principles. The original document was based on inputs from numerous INCOSE contributors, was edited by Joe Defoe, and published in draft form for internal INCOSE review.

INCOSE WMA Chapter
c/o IBM FSC
Mail Stop 409, 6600 Rockledge Drive, Bethesda MD 20817
301-493-1451; FAX 301-493-1542
defoej@ssrc.loral.com

EDITOR'S NOTE

It has been my pleasure to compile this list from the recommendations of the participants. I thank everyone for their participation and apologize for the delay in completing the compilation. The last half of the year was just too busy to give the list the attention it deserved until the free, restful two weeks of the year end holiday season.

My goal was to produce a list that was the union rather than the intersection of the principles that members submitted, thus not everyone will agree that all the items "deserved" to make the list. In editing the text of inputs, and in combining inputs that seemed to address a common issue, I've attempted to achieve a common usage of terms across the lists. I hope this process has not done damage to the spirit of any of the inputs from the participants.

I have not consulted the preexisting literature, preferring instead to have the text reflect the ideas that have accumulated and combined through experience. However, a number of the participants cited Eberhardt Rechtin's *Systems Architecting: Creating & Building Complex systems*, Prentice Hall, 1991, as a source for a large number of excellent heuristics.

[Picture]

Editor's draft subject to change upon review by the contributors.

Document Last Edited 14-Mar-96
Maintained by New Horizons Press -- nhp@fortnet.org

TB Note:

This electronic copy is the most recent and complete that can be found. If anyone has the pictures referenced herein, they are invited to send them to INCOSE Central Office, for addition to the INCOSE Archives and incorporation in this electronic document. This version has been administratively cleaned up by the INCOSE TB editors.

Table of Contents

1	DOCUMENT OVERVIEW.....	6
1.1	BACKGROUND.....	6
1.2	CONTENTS.....	6
2	THE LIST	7
2.1	KNOW THE PROBLEM, THE CUSTOMER, AND THE CONSUMER.....	7
2.2	USE EFFECTIVENESS CRITERIA BASED ON NEEDS TO MAKE SYSTEM DECISIONS	7
2.3	ESTABLISH AND MANAGE REQUIREMENTS	8
2.4	IDENTIFY AND ASSESS ALTERNATIVES SO AS TO CONVERGE ON A SOLUTION	8
2.5	VERIFY AND VALIDATE REQUIREMENTS AND SOLUTION PERFORMANCE	9
2.6	MAINTAIN THE INTEGRITY OF THE SYSTEM.....	9
2.7	USE AN ARTICULATED AND DOCUMENTED PROCESS	9
2.8	MANAGE AGAINST A PLAN.....	10
3	NOTES	10
APPENDIX A	EXISTING DEFINITIONS OF SYSTEMS ENGINEERING.....	12

SE Principles Workgroup

1 DOCUMENT OVERVIEW

1.1 BACKGROUND

This document compiles a set of "pragmatic principles" that underlie the practice of systems engineering. The compilation of the list was begun as a subgroup exercise at the first annual symposium of the National Council on Systems Engineering (NCOSE), October 22, 1991, Chattanooga, Tennessee. Subsequent to the Chattanooga meeting workgroup member comments were consolidated into a draft that was distributed at the January 1992 working meeting and the second annual symposium in July 1992. This final report incorporates the comments received as a result of the January and July distributions.

The starting point for the exercise was the definition of systems engineering contained in the May 15, 1991 Pre-coordination Draft of Mil-STD-499B Systems Engineering:

Systems engineering is an interdisciplinary approach to evolve and verify an integrated and optimally balanced set of product and process designs that satisfy user needs and provide information for management decision making.

Additional definitions of systems engineering are provided for reference in Appendix A.

[Picture]

1.2 CONTENTS

The lists in this document consolidate the contributions from the INCOSE members who participated in the exercise that produced the original list and in the review of the draft document. Each item represents a principle that has either, when heeded, aided the member in practicing successful systems engineering or, when ignored, led to difficulties.

Since the purpose of effort was solely to articulate pragmatic principles, the lists should be viewed as pieces of good advice, system engineering adages, to be taken into account when planning the engineering of a system not as an outline of a complete systems engineering process. In fact, not all the principles apply to all situations.

2 THE LIST

2.1 KNOW THE PROBLEM, THE CUSTOMER, AND THE CONSUMER

1. Become the "customer/consumer advocate/surrogate" throughout development and fielding of the solution.
2. Begin with a validated customer (buyer) need - the problem.
3. State the problem in solution-independent terms.
4. Don't assume that the original statement of the problem is necessarily the best, or even the right one.
5. When confronted with the customer's need, consider what smaller objective(s) is/are key to satisfying the need, and from what larger purpose or mission the need derives; that is, find at the beginning the right level of problem to solve.
6. Determine customer priorities (performance, cost, schedule, risk, etc.).
7. Probe the customer for:
 - ✧ new product ideas
 - ✧ product problem/shortfall identification
 - ✧ problem fixes
8. Work with the customer to identify the consumer (user) groups that will be affected by the system.
9. Use a systematic method for identifying the needs and solution preferences of each consumer group.
10. Don't depend on written specifications and statements of work. Face to face sessions with the different customer/consumer groups are necessary.
11. State as much of the each need in quantified terms as possible. However, important needs for which no accurate or quantified measure exists, still must be explicitly addressed.
12. Clarify each need by identifying the power and limitations of current and projected technology relative to the customer's larger purpose, the environment, and ways of doing business.

2.2 USE EFFECTIVENESS CRITERIA BASED ON NEEDS TO MAKE SYSTEM DECISIONS

1. Select criteria that have demonstrable links to customer/consumer needs and system requirements.
 - ✧ Operational criteria: mission success, technical performance.
 - ✧ Program criteria: cost, schedule, quality, risk.
 - ✧ ILS criteria: failure rate, maintainability, serviceability.
2. Maintain a "need based" balance among the often conflicting criteria.
3. Select criteria that are measureable (objective and quantifiable) and express them in well known, easily understood units. However, important criteria for which no measure seems to exist still must be explicitly addressed.
4. Use tradeoffs to show the customer the performance, cost, schedule, and risk impacts of requirements and solutions variations.

5. Whenever possible, use simulation and experimental design to perform tradeoffs as methods that rely heavily on "engineering judgement" rating scales are more subject to bias and error.
6. Have the customer make all value judgements in tradeoffs.
7. Allow the customer to modify requirements and participate in developing the solution based on the tradeoffs.

2.3 ESTABLISH AND MANAGE REQUIREMENTS

1. Identify and distinguish between specified (fundamental or essential), allocated, implied and derived requirements.
2. Carry analysis and synthesis to at least one level broader and deeper than seems necessary before settling on requirements and solutions at any given level. (Top-down is a better recording technique than it is an analysis or synthesis technique.)
3. Write a rationale for each requirement. The attempt to write a rationale for a "requirement" often uncovers the real requirement.
4. Ensure the customer and consumer understand and accept all the requirements.
5. Explicitly identify and control all the external interfaces the system will have - signal, data, power, mechanical, parasitic, etc. Do the same for all the internal interfaces created by the solution.
6. Negotiate interfaces with affected engineering staff on both sides of each interface and get written agreement by the two parties before the customer approves the interface documentation.
7. Document all requirements interpretations in writing. Don't count on verbal agreements to stand the test of time.
8. Plan for the inevitable need to correct and change requirements as insight into the need and the "best" solution grows during development.
9. Be careful of new fundamental requirements coming in after the program is underway. They invariably have a larger impact than is obvious.
10. Maintain requirements traceability.

2.4 IDENTIFY AND ASSESS ALTERNATIVES SO AS TO CONVERGE ON A SOLUTION

1. Take the time to innovate by generating a wide range of alternative solutions to satisfy the need. (A common mistake is to converge on a "comfortable design" concept too early because of time constraints.)
 - ✧ Consideration of seemingly bizarre alternatives often yields additional insight into the requirements and provides a reasonableness check for tradeoff criteria and weights.
 - ✧ Include the "do nothing solution" in the system level solution tradeoff to provide a measure of the value-add the new system will bring the customer/consumers.
2. Use a systematic architecture/design method.
 - ✧ Abstract the requirements to identify the essential design problems.
 - ✧ Establish functional structures.
 - ✧ Search for solution principles to fulfill the sub-functions.
 - ✧ Combine the solution principles to fulfill the overall functions.

- ✧ Select suitable combinations.
- ✧ Firm combinations into conceptual alternatives.
- 3. Evaluate each alternative against the requirements and the effectiveness criteria. Determine the alternative that provides the best weighted value combining:
 - ✧ effective
 - ✧ efficient
 - ✧ safe
 - ✧ reliable
 - ✧ producible
 - ✧ testable
 - ✧ maintainable
 - ✧ easiest to learn
- 4. Elaborate the customer's top-level concept of operations to show how the consumers will use each solution alternative to satisfy the consumers' and the the customer's needs. This detailed concept of operations must be reflective of the design aspects of the system's operation.

2.5 VERIFY AND VALIDATE REQUIREMENTS AND SOLUTION PERFORMANCE

1. Quality must be designed in, it cannot be tested in.
2. Use preplanned peer reviews and inspections.
3. Prototype critical elements.
4. Use models to demonstrate feasibility before bending metal and writing code.
5. Explicitly indentify and santity check all model assumptions.
6. Work the critical and controversial requirements and design areas first.
7. Plan the verification and validation for every requirement.
8. Know the expected results before testing.

2.6 MAINTAIN THE INTEGRITY OF THE SYSTEM

1. Maintain a systems engineering presence throughout the program (even though SE staff starts to drop off after PDR and more after CDR) to provide technical oversight of the ongoing design process and to resolve requirements/technical issues that invariably arise, including resolution of test discrepancies/anomalies.
2. Prevent process and product contamination.
3. Ensure the system design meets the requirements, satisfies the need, and reflects the voice of the customer.
4. Ensure the requirements address not only the operational objectives but all the life-cycle objectives for the system.

2.7 USE AN ARTICULATED AND DOCUMENTED PROCESS

1. Start with established principles - avoid reinventing the wheel and really learn from "lessons learned" investigations.
2. Use the principles to develop a process tailored to the need, the system, the customer, and the development organization.

Use or disclosure of information contained herein is subject to the restrictions on the title page of this document
INCOSE Technical Information

3. Use the process consistently across the program.
4. Train the development staff in the process and its application - technical education is one key to productivity, quality, and cost reduction.
5. Use standardized analysis techniques, document formats, design review formats, etc. to reinforce the consistent application of the process.
6. Use readily available automated tools wherever appropriate.
7. Maintain process integrity but never let the process prevent the "best" solution from being discovered or used - do whatever it takes to build in product quality.

2.8 MANAGE AGAINST A PLAN

1. Use a "tasks are executed to produce useful work products" focus for the plan.
2. Prepare a plan that is success oriented, achievable, defensible, and cost-effective but which can absorb the changes that will come.
3. Have a contingency plan for each identified risk.
4. Develop a plan that reflects organizational commitment to systems engineering.
5. Look for and abolish fraction-of-a-job situations.
6. Perform each task according to the plan.
7. Change the plan as soon as experience shows a better way to do a task.
8. Remember: micromanagement is not planning.
9. Remember Dwight D. Eisenhower's words: "Plans are nothing. Planning is everything."

[Picture]

3 NOTES

(1) This definition has been revised over the years while its essence has remained. The sources used in constructing it back in 1983 are shrouded by the mists of time and the retirement of the original headquarters team who first published it.

(2) "Build: To give form to according to a definite plan or process; create." *American Heritage Dictionary*, Second College Edition

(3) This definition cited by Aaron Shenhar ("Developing the Discipline of Systems Engineering," INCOSE/ASEM Proceedings October 1991, ASEM-148,149) in his discussion of the definition of systems engineering. This paper also identifies other definitions and finds the following common elements in them:

1. It directs the design, development, synthesis and creation of a system, rather than the analysis of it, and the transforming of an operational need or a customer's need into an existing systems
2. It involves a "holistic view" (the design of the whole as distinguished from the design of the parts). Such a view is multidisciplinary in nature, rather than disciplinary or interdisciplinary.

3. It involves a technical part and a managerial part; namely, it requires making technical decisions and trade-offs while controlling and managing the efforts of different experts and teams from various disciplines.

NHP/mao / 20-Apr-95 / nhp@fortnet.org

APPENDIX A EXISTING DEFINITIONS OF SYSTEMS ENGINEERING

Mil-STD-499B Systems Engineering

Systems engineering is an interdisciplinary approach to evolve and verify an integrated and optimally balanced set of product and process designs that satisfy user needs and provide information for management decision making.

MIL-STD-499A Systems Engineering

... the application of scientific and engineering efforts to

- transform an operational need into a description of system performance parameters and a system configuration through the use of an iterative process of definition, synthesis, analysis, design, test, and evaluation;
- integrate related technical parameters and ensure compatibility of all physical, functional, and program interfaces in a manner that optimizes the total system definition and design;
- integrate reliability, maintainability, safety, survivability, human engineering, and other such factors into the total engineering effort to meet cost, schedule, supportability, and technical performance objectives.

DSMC Systems Engineering Management Guide, January 1990, pg 1-2.

... is the management function which controls the total system development effort for the purpose of achieving an optimum balance of all system elements. It is a process which transforms an operational need into a description of system parameters and integrates those parameters to optimize the overall system effectiveness.

IBM Federal Systems Company definition used as the basis for divisions, internal SE practices and education. (1)

The iterative controlled process in which users needs are understood and evolved, through incremental development of requirements specifications and system design, to an operational system.

Joe DeFoe and Jim McAuley; November 1991

The processing of building (2) real things to solve real problems within technological, environmental, economic, legal, ethical, cultural, and institutional constraints.

Computer Aided Systems Engineering, Howard Eisner, Prentice Hall, 1988, pg 17.

... an iterative process of top-down synthesis, development, and operation of a real-world system that satisfies, in a near-optimal manner, the full range of requirements for the system.

Field Manual: System Engineering, FM 770-78 Headquarters, Department of the Army, April 1979, pg 1-2. Summary definition in the introduction of the manual.

The transforming of an operational need into a description of system performance parameters and a system configuration.

Field Manual: System Engineering, FM 770-78 Headquarters, Department of the Army, April 1979, pg C-2. More extensive definition found in the glossary.

The selective application of scientific and engineering efforts to:

1. transform an operational need into a description system configuration which best satisfies the operational need according to the measures of effectiveness;
2. integrate related technical parameters and ensure compatibility of all physical, functional, and technical program interfaces in a manner which optimizes the total system definition and design;
3. integrate the efforts of all engineering disciplines and specialties into the total engineering effort.

"Fundamentals of Systems Engineering at NASA," INCOSE/ASEM Proceedings, October 1991, Robert G. Chamberlain and Robert Shishko, Phd., pg 23.

... is a robust approach to the design and creation of systems to accomplish desired ends.

P. M'Pherson, "Systems engineering: A proposed definition," IEE Proce., Vol. 133, pp. 330-331, Sep. 1986 (3)

... is a hybrid methodology that combines policy analysis, design and management. It aims to ensure that a complex man-made system, selected from the range of options on offer, is the one most likely to satisfy the owner's objectives in the context of long-term future operational or market environments.

[Picture]